KU LEUVEN

FACULTY OF ECONOMICS AND BUSINESS

Matching bibliographic data from publication lists with large databases using N-Grams

Mehmet Ali Abdulhayoglu Bart Thijs and Wouter Jeuris

Matching Bibliographic Data from Publication Lists with Large Databases using N-Grams

Mehmet Ali Abdulhayoglu, Bart Thijs, and Wouter Jeuris

KU Leuven

Author Note

Bart Thijs, Centre for R&D Monitoring (ECOOM) and Dept. MSI, KU Leuven, Waaistraat 6, Bus 3536, Leuven, Belgium. e-mail: Bart.Thijs@kuleuven.be

Wouter Jeuris, Centre for R&D Monitoring (ECOOM) and Dept. MSI, KU Leuven, Waaistraat 6, Bus 3536, Leuven, Belgium. e-mail: Wouter.Jeuris@kuleuven.be

Correspondence concerning this article should be addressed to Mehmet Ali Abdulhayoglu, Centre for R&D Monitoring (ECOOM) and Dept. MSI, KU Leuven, Waaistraat 6, Bus 3536, Leuven, Belgium. Contact: Mehmetali.Abdulhayoglu@kuleuven.be

Abstract

This paper presents a text matching process for identification and correct assignment of scholarly publications, extracted from publication lists provided by authors or research institutes, in large bibliographic databases such as Thomson Reuters' Web of Science (WoS). An identification method is implemented by means of overlapping common 3-grams and the results are obtained from the match of the two sources according to the highest score of the applied cosine measure. Levenshtein similarities based on N-grams have been used to measure the closeness between the given CV publication and the retrieved best possible WoS match as a complementary and confirmatory measure. It is shown that the suggested method has an important potential on reducing the manual effort to find out whether a desired publication is indexed in WoS or not. The similarity scores derived by Levenshtein measure show consistency with those derived from Salton's similarity measure. Incorrect matches are examined in depth and possible thresholds are suggested to decrease the effort for manual cleaning.

Keywords: string matching, n-gram, edit distance, levenshtein distance, information retrieval

Matching Bibliographic Data from Publication Lists with Large Databases using N-Grams

With the rapid, enormous growth of scientific literature, which is partially due to the increasing team work and the growing national and international collaboration and globalisation, bibliometric evaluation has become increasingly important and as a supplementary task to peer review and expert opinion even more complex as well. In the light of these challenges, bibliometrics more and more evolved from a discipline analysing and modelling the scientific communication to a tool for benchmarking and evaluating research performance (Glänzel and Schoepflin, 1994; van Raan, 1997).

One of the tasks of the field as a tool for evaluating research performance is retrieving publications in large and prominent databases like Thomson Reuters' Web of Science (WoS) and Elsevier's Scopus. Therefore, the application of evaluative bibliometrics, especially at the micro and meso level often requires the use of individual CVs and institutional publication lists as an input of the analysis. The retrieved data sets are often a crucial determinant of the validity of the final results. Therefore, the identification of the authors' or research teams' publications in large databases usually requires a tremendous amount of manual effort. Automation of this process of directly linking publications provided in lists to publication records indexed in the database could essentially simplify this task and free up resources that had otherwise been assigned to the manual cleaning tasks.

Most commonly, the problem in finding such matches is caused by incomplete, erroneous or censored data in publication lists; but erroneous entries occur in the databases as well. It is very likely that an author adds a publication to his/her CV as soon as a paper is submitted or conditionally accepted but later, the actual publication year, but also the title or the number and sequence of co-authors can change during the process of revision and finalization of the publication.

In the present paper, a promising retrieval method based on character N-grams is presented. In the literature, there are numerous works related to reference/citation matching for many different purposes (e.g. Giles et al., 1998; Lawrence et al., 1999; Larsen, 2004; Piskorski and Sydow, 2007).

In a first step, information provided in the CV records (e.g., article and journal title, name of co-authors, etc.) is treated as queries to be searched within whole bibliographic database. To find the most likely matching reference for each CV publication, Salton's cosine measure (Glänzel and Czerwon, 1996) based on overlapping unique N-grams (bag of unique N-grams) is applied.

In the second phase, a more elaborate and resource intensive similarity measure is used to confirm and complement the cosine measure. This method was introduced by Kondrak (2005) and is a modified Levenshtein (edit) distance based on N-grams. Since the order (position) of the N-grams matters when using Levenshtein distance, some more steps have to be taken to perform the comparison and later on the assignment. For the pairs collected in the first step, similarity scores are derived based on this method and the similarity scores for each pair are then examined.

The objective of this study is not to completely replace manual work by an automated process – a task that would in fact be unrealistic, but to reduce manual work to a reasonable minimum by providing the most likely matches for as many publications as possible.

In order to test the efficiency of the proposed method, a publication list taken from CVs provided by a set of scientists has been matched with papers indexed in the WoS database. The results show that the suggested model is capable of retrieving the publications in our set with almost 99% accuracy when only the highest Salton similarity scores of the pairs are retained.

In the remaining part of this paper, we introduce the concept of character N-grams with its history and we discuss why they are more superior over word N-grams in our application by giving their advantages. Then the calculation of Salton's cosine measure and Levenshtein distance based on N-grams is explained in detail. This is followed by the explanation of the retrieval process and application of Levenshtein distance. Finally, data is described and results are discussed in the concluding sections.

Methodology

N-grams

A (word) character N-gram is an adjacent sequence of *n* (words) characters from a given text. For instance, *sea ear arc rch ch_ h_p _pr pro roj oje jec ect* are the character 3-gram fragments of the string *search project* and *"bibliometric and scientometric" "and scientometric research" "scientometric research project"* are the word 3-gram partitions of the text *"bibliometric and scientometric research project"*. The history of character N-gram, as our main focus, dates back to 1948, when it was described by Claude Shannon. The application of this approach emerged from the need of decreasing the size of dictionaries. In other words, number of words contained in a collection is theoretically infinite as the collection grows whereas the number of distinct character N-grams, especially for the small *n*, stays finite (e.g. there are at most 27^3 3-grams for the English alphabet of 26 letters and space, cf., McNamee and Mayfield, 2004).

Considering this advantage, many studies were performed on the efficiency of (short) character N-grams in the field of information retrieval (Damashek, 1995; Cavnar, 1994; Comlekoglu, 1990; Teufel, 1988). Besides information retrieval, the character N-gram approach has also been used in different areas such as language identification (Cavnar and Trenkle, 1994), spelling error detection (Zamora et al., 1981), keyword highlighting (Cohen,

1995), restoration of diacritical marks (Mihalcea and Nastase, 2002), detection of malicious codes (Abou-Assaleh et al., 2004) and anti-spam filtering (Kanaris et al., 2007).

There are many advantages of using character N-grams instead of word-based N-grams. First of all, they do not require pre-processing and are language independent, unlike word based N-grams (Kešelj et al., 2003; McNamee and Mayfield, 2004). The character N-gram technique does not require stemming either, since the corresponding forms of a word (e.g. 'search', 'searching', 'searched') have much in common when decomposed into N-grams (Cavnar and Trenkle, 1994). All methods dealing with words require stemming, parsing, lemmatisation, making use of stop-word lists, phrase lists, lexica, thesauri etc. for obtaining satisfactory results in the retrieval. These pre-processing steps are quite language specific, for instance, a parser designed for English cannot be applied to German or Chinese (Kešelj et al., 2003). Secondly, textual errors or spelling variations can be handled effectively using character N-grams, since each string is decomposed into fragments of text. If an error occurs, this is only included in a small number of N-grams and all other pieces remain intact. At the same time they are also able to cope with abbreviations. That is, when comparing a word with its abbreviation (e.g. *professor* vs. *prof.*), in contrast to word-based systems, common N-grams will keep the similarity between them. Thirdly, the number of distinct N-grams is restricted by the combinations of letters in an alphabet, while the number of words in a collection is infinite as the collection grows. When new publications are added to a collection, many new words might show up, resulting in a permanent inflation. Therefore, in many systems using word based N-grams, words are indexed according to the user's needs. That is, not all words are indexed because of unbounded size and this may cause a drawback in case of a change in the focus of the system that might be required (Adams and Meltzer, 1993).

Despite its notable advantages, character N-grams result in an increase of the size of the inverted index dealt with: a string consisting of *k* letters has at least *k-n+1* number of N-grams with a length of *n* and this number can be increased by adding blank prefixes to the string in order to increase the importance of the initial letter as Kondrak (2005) describes. However, the number of words that may be found in that string would be less. This dimension issue is the most challenging part in terms of storage and performance. Additionally, they are not capable of coping with homographs and might be just as ambiguous as word N-grams are. Nevertheless, if many common character N-grams are found in two strings, this can compensate for this ambiguity issue (McNamee and Mayfield, 2004).

In the light of these pros and cons, we decided to use character N-grams as being suitable for our purposes. The publications in the provided publication lists and in WoS, are not necessarily in English. In addition, abbreviations and textual errors are very likely to be found. Considering the fact that normalized (processed) word-based systems are not cheap either (Adams and Meltzer, 1993) and that character N-grams are superior over un-normalized (raw) words in terms of accuracy (McNamee and Mayfield, 2004), we implement the character N-gram approach to our given unassisted data without any pre-processing.

In the literature, a number of studies investigating the size of *n* in terms of suitability for information retrieval and classification (Cavnar and Trenkle, 1994; McNamee and Mayfield, 2004; Comlekoglu, 1990) can be found. In the present study we decided to use 3-grams since the length of the components forming the bibliographic information is rather short. That is, since such components as author names or publication year comprise short texts, there is no need to use N-grams in big sizes to grab a straightforward similarity. In addition, indexes created with character 3-grams are small enough to be memory resident independently of the size of the database. This is a clear advantage to word-based indexes

that are unbounded and need to be kept on disk (Adams and Meltzer, 1993). Lastly, character N-grams provide the connections between words to improve phrase matching (Cavnar, 1993).

Salton Cosine Measure based on N-grams

In the bibliometrics literature, *"coupling angle"* is frequently applied when a measure of relationship is based on a Boolean vector-space model (Sen and Gan, 1983). In our study, the Boolean vector space is represented by the relationship between publications from two different sources (CV publications and WoS publications) and the set of all unique 3-grams included by them. Sen and Gan define the measure as the cosine of two Boolean vectors.

Glänzel and Czerwon (1996) have shown that the coupling angle is identical with Salton's measure in such a comparison of Boolean vectors. As given in Eq. (1), we can define Salton's measure as the ratio of the number of joint unique 3-grams and the geometric mean of the number of unique 3-grams from two papers (recorded in the CV and indexed in the WoS database, respectively). Salton's measure takes the value 1 if two compared vectors are identical and 0 if they are completely different, i.e., if they do not share any 3-grams.

$$sim = \frac{c}{\sqrt{a * b}} \tag{1}$$

where,

a: total number of unique 3-grams of a record from publication list

b: total number of unique 3-grams of WoS publication

c: total number of common unique 3-grams

A more detailed explanation about the calculation process will be depicted in the information retrieval section (*Phase 1*).

Levenshtein (Edit) metrics based on character N-grams

Levenshtein distance is a way of measuring the dissimilarity (or, alternatively the similarity as) of two strings by counting the minimum number of single-character edits

required to transform one string to the other (Levenshtein, 1966). In the Levenshtein distance, insertion, removal or substitution of a single character are allowed operations. The idea introduced by Kondrak (2005) is to use character N-grams instead of single letters when measuring edit distance.

In essence, when applying N-gram based edit distance for strings *x* and *y* with a length of *a* and *b* respectively, a matrix $M_{1\ldots a+1,1\ldots b+1}$ is constructed, where $M_{i,j}$ is the minimum number of edit operations needed to convert $x_{1\ldots i}$ to $y_{1\ldots j}$ or vice versa. Each matrix element $M_{i,j}$ is calculated according to Eqs (2) – (4), where the 'cost' in equation (3) is the total number of times that letters in the same positions are distinct in the N-grams x_i, y_j, and n is the size of N-gram.

$$M_{1,1} = 0 \tag{2}$$

$$M_{i,j} = min\begin{cases} M_{i-1,j} + 1 \\ M_{i,j-1} + 1 \\ M_{i-1,j-1} + \delta(x_i, y_j) \end{cases} \tag{3}$$

$$\delta(x_i, y_j) = cost/n \tag{4}$$

Figure 1 presents a simple application of Levenshtein distance based on 3-grams between the two strings *'diffusion'* and *'diff.'*. The circled cells in the figure are the related cells to measure the minimum edit distance between sub-strings *'diff.'* and *'diffu'*, which is given in cell (R6, C6). According to (3), 1 is added for insertion and 1 for removal operations to the previous minimum distances, which are 1 (insertion) to convert *'diff*' to *'diffu'* given in cell (R5, C6), and 1 (removal) to convert *'diff.'* to *'diff*' given in cell (R6, C5). For the substitution operation, $\delta(x_i, y_j)$ is added to the previous minimum distance 0 (between *'diff*' and *'diff*') given in cell (R5, C5)$\delta(x_i, y_j)$. Here the related 3-grams are *'ffu'* and *'ff.'* for the calculation of $\delta(x_i, y_j)$. In Eq. (4), cost is the number of change operations needed to make the two 3-grams congruent. In our example, cost is 1 since the letters appearing in the first two orders are identical and by dividing this cost by the size of the N-gram, which is 3,

$\delta(x_i, y_j)$ is obtained with value 0.33 according to Eq. (4). Finally, the minimum distance for the specific cell is obtained by taking the minimum among the calculated distances (2, 2, 0.33). For each cell the related minimum distance is calculated and the last cell of the matrix given in cell (R6, C10) gives the final distance (4.33) between two strings.

"diffusion" string and its 3-grams

"diff" string and its 3-grams			C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
				d	i	f	f	u	s	i	o	n
				__d	_di	dif	iff	ffu	fus	usi	sio	ion
R1			0	1	2	3	4	5	6	7	8	9
R2	d	__d	1	0	1	2	3	4	5	6	7	8
R3	i	_di	2	1	0	1	2	3	4	5	6	7
R4	f	dif	3	2	1	0	1	2	3	4	5	6
R5	f	iff	4	3	2	1	0	1	2	3	4	5
R6	.	ff.	5	4	3	2	1	0.33	1.33	2.33	3.33	4.33

Figure 1. N-gram based Levenshtein distance example.

The suggested Levenshtein approach normalizes the final distance measure by the length of the larger string to avoid length bias. In addition, it adds a null-character prefix of size *n*-1 so that the initial letter can be exploited more efficiently since the initial characters play an important role in word similarity (Kondrak, 2005). It should be stressed that null-character prefix matches are discarded so that strings with no matching characters will return maximum distance. Finally, distances are subtracted from 1 to get the similarity scores ranging between 0 and 1, where 1 or 0 means that the specified strings are identical or completely different in terms of 3-gram representation, respectively.

From the description and its properties, it is clear that the suggested Levenshtein distance is sensitive to the positions of the N-grams in strings, unlike the traditional N-gram approach, which considers only the number of overlapping unique N-grams. When a high similarity score between two strings is obtained from the Levenshtein distance, this similarity score might be more reliable than the one derived from the traditional approach since the Levenshtein distance not only takes the common N-grams into account but also their position within the strings. In order to take full advantage of this feature, some possible reference combinations are created by means of the indexed components in WoS and these references

are compared with the given CV publications. These possible reference variants need to be built since we are using CV publication records that follow unknown reference standard or might even have incomplete components. As a result, the components (author, journal, title etc.) may be given in an arbitrary order, or some parts may be missing. This issue is explained in depth in the context of the second step (*Phase 2*).

Phase 1 – Database retrieval of the most probable matches for the given CV publications

In this section, the main objective is to find the most relevant publications indexed in Thomson Reuters' WoS for a given publication list, in this case a CV provided by an individual author. The retrieval process is shown in Figure 2. In a usual document retrieval system, the user specifies the necessary information using small queries (Goller et al., 2000). In our study, we use the unique 3-grams extracted from the records in the publication lists as queries to be searched for through WoS.

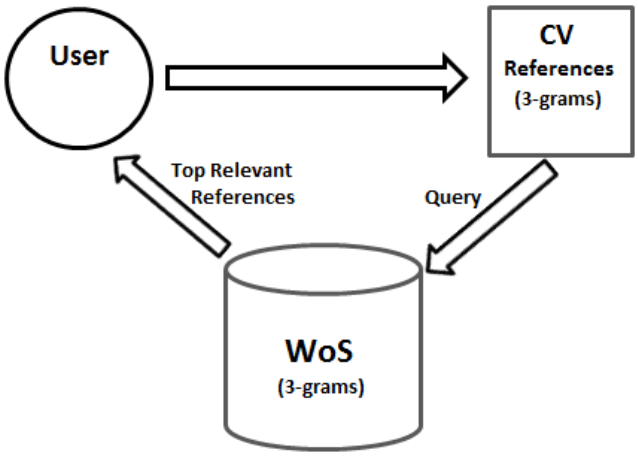

Figure 2. Retrieval process.

The given set contains many publications from different authors and the format used for the references in the CVs is unknown and varied. They may contain textual errors, spelling variations and abbreviations. In addition, their components (title, journal name, authors etc.) might be placed in an unknown order and some components might be missing or

incomplete, or unknown and unnecessary components might be present. Considering these challenges, our aim is to extract an entirely unassisted data set from WoS.

As mentioned in the *N-grams* section above, using 3 as the size of N-grams is useful for finding similar publications in an unassisted way. Furthermore, memory capacity might be a crucial technical issue, which might create a bottleneck when dealing with such huge data sets. Comlekoglu (1990) investigated N-gram inverted text retrieval systems for n=2 to n =26 and he found that when main memory was scarce, 3-grams were a reasonable choice in designing an N-gram text retrieval system.

Before getting started with the retrieval process, we derive unique 3-grams from the components title, journal name, journal volume, co-author names, publication year and first-page number, which are expected to appear in the bibliographic items for each publication existing in WoS. For proceeding papers, also conference information is added. Then decomposition into 3-grams is applied to all the publications in the CVs. All 3-grams, which do not exist in the publication lists but do exist in WoS, are removed from the WoS set and the opposite of this case is treated similarly. Indexes are then assigned to the remaining 3-grams that appear in WoS, as well as in the CV. In addition, indexes are created for each publication in the WoS and each CV publication and all strings in both corpora are transformed to lowercase.

According to the unique 3-grams that the documents (WoS or CV publications) contain, unique feature (3-gram) vectors are created. A feature vector represents a document with the weights of features (Goller et al., 2000). In our study, we only apply binary weights, that is, the information whether a feature occurs in a document or not. Another weighting approach such as the TF-IDF might reflect how important a feature is for a document (Salton and Mc Gill, 1983) and might have a significant potential of improving retrieval and classification processes. However, we are working on short texts most likely having only a

few 3-grams occurring more than once, which results in an inappropriate TF-IDF weighting scheme (Gong et al., 2008).

On the basis of the feature vectors two sparse matrices are created using MATLAB 2012b, one each for WoS papers and the CV publications. These matrices and the following matching process are shown in Figure 3. In these matrices, value 1 indicates that the 3-gram in the relevant column exists in the document presented by the relevant row. As seen, the column order (3-gram indexes) has to be identical in both matrices.

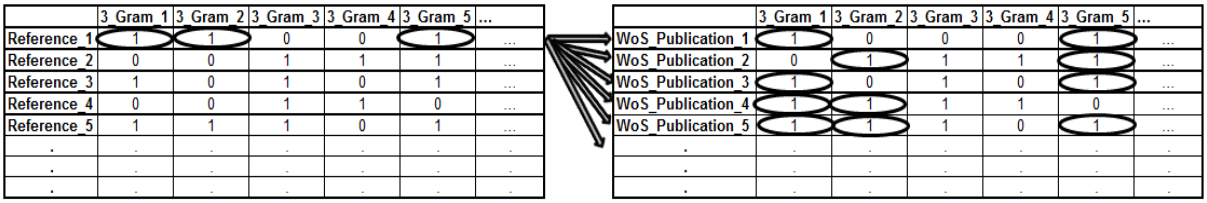

Figure 3. Matching process of CV publications with WoS Publications according to their overlapping unique 3-grams.

During the matching process, the column positions of the values 1 from the first row of the *CV* publication matrix are taken and the same column positions are chosen for each of the row from the *WoS* matrix. The total number of common values 1 gives the number of common 3-grams for each WoS and CV publication pair. Thus, for instance, the positions of values 1 in the *CV* publication matrix for the first row in Figure 3 are the 1st,2nd and 5th. In the *WoS* matrix, only these three positions are retained. According to these matches, the WoS publication in the first row has 2 common 3-grams and that in the second row has 2 common 3-grams and so on. Using these common 3-grams, Salton's measure is calculated as given before for each pair. This process is repeated for each CV publication one by one from the publication lists set and the most relevant WoS publication according to Salton's measure are retained for each CV record.

Here it should be mentioned why we make use of only the columns including the values 1 for each row from the *CV* publication matrix and multiplying it with the entire *WoS*

matrix with its corresponding columns instead of simply multiplying two matrices. The two matrices are sparse matrices and the multiplication of two sparse matrices in MATLAB results in a very slow processing. In order to parallelize the matrix multiplication, the matrices have to be converted into dense matrices. However, this conversion process is very likely to give out-of-memory exception especially for huge matrices such as our *WoS* matrix. By taking certain columns from *WoS* matrix depending on the row of *CV* publication matrix having only the columns with values 1s, we reduce the dimension of *WoS* matrix and make it possible to convert it to a dense matrix. In our case the dimension of the *WoS* matrix was 1.793.835×20.615 for the year 2007 and could be reduced to 1.793.835×500 according to the given CV publication).

Phase 2 – Application of Kondrak's Levenshtein distance

In this phase, the most relevant WoS publication retrieved in the previous phase for each CV publication is re-matched with the related CV publication and the similarity scores are derived according to Kondrak's Levenshtein distance based on N-grams. We just mention in passing that this can easily be implemented, for instance, through the *'NGramDistance'* class included in *LUCENE*, a high-performance text search engine library written in *Java*.

Unlike in the previous approach, where all components, that are likely to appear in the bibliographic information of a given publication, are concatenated and unique 3-grams are formed, this time all 3-grams and their positions in the components are taken into account. Using this metric, the closer the identical N-grams appear in the reference publication, the higher the similarity score will be.

Publication lists provide detailed bibliographic information about the publications such as title, journal title, co-author names, journal volume, first page, etc. However, for any given record, the order or formatting of these components might change depending on the reference standard used (e.g. American Psychological Association (APA), Modern Language

Association (MLA), non-standard reporting, etc.). For instance, while co-author names might appear in some references with full names, they might appear with only last name and initial letter of the first and/or intermediate name. And as we already mentioned, the order of the components might be different in the references. Other deviations might result from mistakes made by the author while creating the list, or corrupted characters and missing components resulting from transferring the information between different electronic environments.

On the part of the WoS, the components for each publication are stored in separate tables in a relational database. Departing from this fact, various combinations are constructed with the related components in different orders. In this manner, we derive various similarity scores for each pair of CV publications and its related WoS counterparts and obtain that WoS combination with the highest score.

Table 1 contains an overview of the constructed variations. In each row, the numbers in the table indicate the order of the components used to construct the references from WoS records. Note that not all components are always used. For each variation, the corresponding similarity score for the associated CV publication is calculated and stored in variables Score1 to Score8.

Table 1

Structured References by means of the Components in Database

Variables	WoS Title	WoS Journal	WoS Co-authors	WoS Volume	WoS Begin Page	WoS Publication Year
Score1	2	3	1	4	5	6
Score2	2	-	1	3	4	5
Score3	1	2	-	3	4	5
Score4	1	-	2	-	-	-
Score5	1	-	-	-	-	-
Score6	1	2	-	-	-	-
Score7	-	1	2	-	-	-
Score8	-	1	-	-	-	-

Table 2 shows an example of 3-gram similarity scores. One entry from a CV publication list is matched with six different representatives of the same WoS publication.

Table 2

CV and Variable Similarity Scores with 3-grams [Data sourced from Thomson Reuters Web of Knowledge]

Source	CV Reference and Its Corresponding Structured WoS References	Variables	3-Gram Score
PL ref.	Zhang, L., Thijs, B., Glänzel, W., The diffusion of H-related literature. JOI, 2011, 5, 583-593		
WoS	ZHANG, L, THIJS, B, GLANZEL, W, The diffusion of H-related literature, JOURNAL OF INFORMETRICS, 5, 583, 2011	Score1	.67
WoS	ZHANG, L, THIJS, B, GLANZEL, W, The diffusion of H-related literature, 5, 583, 2011	**Score2**	**.73**
WoS	The diffusion of H-related literature, JOURNAL OF INFORMETRICS, 5, 583, 2011	Score3	.34
WoS	ZHANG, L, THIJS, B, GLANZEL, W, The diffusion of H-related literature	Score4	.65
WoS	The diffusion of H-related literature	Score5	.36
WoS	The diffusion of H-related literature, JOURNAL OF INFORMETRICS	Score6	.41
		Maximum	**.73**

At this point, it should be mentioned that the application of this approach to a huge database is not feasible with the current available technology and there are two causes for this. Firstly, this approach uses all the N-grams, not just the common N-grams, and also includes their positions leading to a significant increase of dimensionality. Secondly, various similarity scores are calculated by means of different component combinations to catch the highest scores for each CV-WoS pair. In the current approach, each CV reference has only one related WoS publication namely that with the highest Salton similarity score. When applying this to the complete database, it would have millions. Even deriving one similarity score is time- and storage-consuming, notably for retrieving the similarity scores for 8 different representations.

Aware of the sensitivity of this measure to N-gram positions, we expect to derive more realistic similarity scores. A closer analysis might reveal that two strings with many common N-grams might not be so similar in fact. However, if one can find two strings having many N-grams in common at the same positions, they are more likely to stand for the same publication. If the similarity scores derived by Salton's measure mirror those obtained from Levenshtein distance, Kondrak's method might be a suitable tool for validating and confirming the results. On the other hand, if inconsistencies are observed between the scores based on these two measures, Kondrak's method might be used as a supplementary measure again.

Data Sources

A large data sample comprised of 6.520 CV publication references mentioned in CVs of various authors has been collected from a real world application in order to find the same, or the most likely related publications indexed in the WoS database. In essence, the exact matches are already known through a manual process and the percentage of the retrieval success is investigated by examining the top Salton matches. Our WoS database contains 28.269.653 articles and 7.044.304 proceeding papers in the period 1991–2012.

The CV and WoS publication datasets have each at most 20.776 unique 3-grams. Number of total unique 3-grams changes depending on the WoS publication year in which CV publications are searched. For instance, the dimension of the sparse CV publication matrix shown in Figure 3 was 6.511×20.615 when it was created for search in the 2007 volume of the WoS. On the other hand, the WoS matrix was huge and sparse, in particular, 1.793.835×20.615 for the year 2007. After retrieving the top matches, the same 6.520 CV publications with their corresponding WoS publications was used to apply the Levenshtein distance method.

Results and Discussion

In the retrieval phase, we managed to match CV publications correctly to WoS publications with a very high accuracy. Each CV publication is matched with a WoS publication according to the highest Salton score. Out of 6520 matched pairs, 6493 pairs have been matched correctly according to their highest similarity score. This corresponds to an accuracy of 99,58%. The remaining 27 incorrect matches are analysed in depth below.

Through the retrieval method using the Salton measure we took one WoS publication for each CV publication according to the highest Salton score. As a result, we make each CV publication ready for being used in a more elaborated application, namely N-gram based Levenshtein distance. Consequently, another similarity score was derived to check and confirm the accuracy of the Salton measure.

The two diagrams of Figure 4 visualizes the top related CV–WoS matches. The upper one (Figure 4a) shows the top related CV–WoS publication pairs having the highest Salton score (*x-axis)* and their corresponding similarity score based on Levenshtein distance (*y-axis)*. The circles in the figure represent correct matches while the squares stand for the incorrect ones. In order to improve legibility, these sparse squares are presented in the lower diagram (Figure 4b). They are also grouped into four different classes according to their types. These will be discussed in detail later in this section.

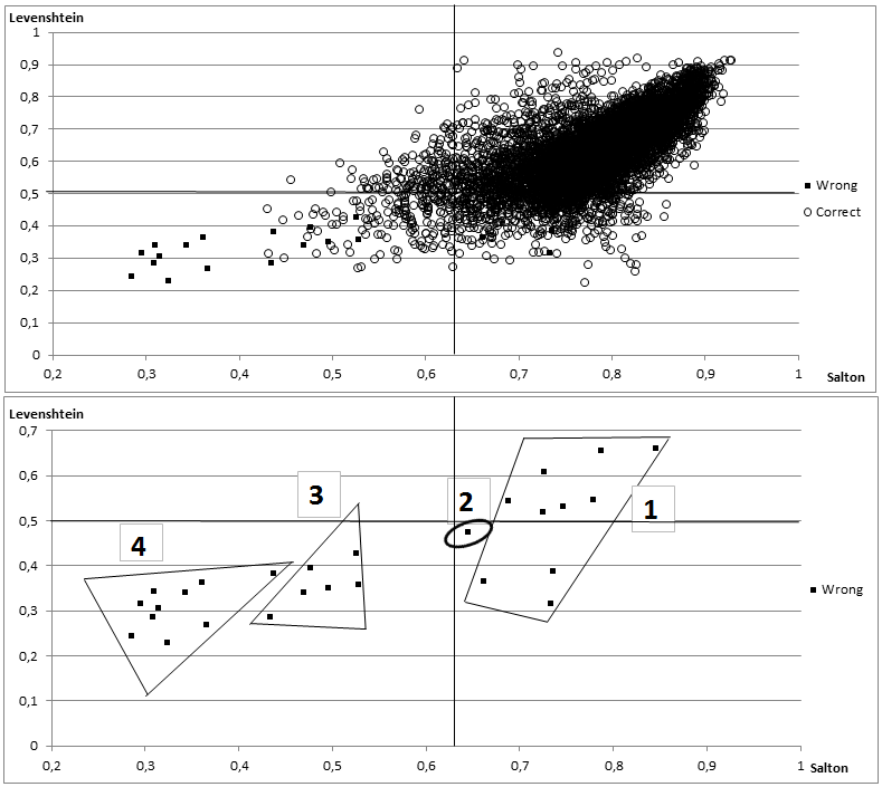

Figure 4. The top CV publication-WoS publication matches according to the highest similarity score based on Salton measure and the corresponding Levenshtein score (Upper) (Figure 4a) and All wrong matches (Lower) (Figure 4b) partitioned into groups according to their relationships. [Data sourced from Thomson Reuters Web of Knowledge]

At a first sight, Salton measure and Levenshtein measure give consistent results. That is, if a similarity score based on Salton is high (low), then the similarity score based on Levenshtein is also mostly high (low). Nevertheless, there are still some pairs having slightly different Salton scores when compared with Levenshtein scores (cf. Figure 4a).

As we analyse the possible differences between two measures, lower Salton than Levenshtein score occurs when a CV publication does not include components that are present in WoS and the correct component orders given in the CV can be grabbed through one of the variables we construct for the Levenshtein measure (see Table 1). Another case (higher Salton than Levenshtein score) is mostly found when a CV publication contains components that are not present in the references we have constructed based on the WoS publications or

when a paper has a huge number of co-authors. Too many co-authors increase the risk of reporting c-authorship in a different order than in the corresponding WoS reference. This, in turn, might lead to a smaller Levenshtein score while it will not dramatically affect the Salton score since this only deals with common N-grams regardless of their positions.

Figure 4a visualizes that Salton's measure tends to give higher similarity scores compared to the Levenshtein distance. Even though the Levenshtein distance has some sensitive issues as mentioned above, it might give a more balanced similarity score when the correct order of the components given in the CV is reached also by the constructed WoS reference. Therefore, applying both measure together will be more robust.

As already mentioned, more than 99% of the correct CV–WoS pairs can be retrieved by taking the highest Salton similarity score. Despite a very small number of wrong matches, there are still some pairs having quite high similarity scores according to both measures, which should be further examined.

First of all, a meeting abstract, a correction, a part, a reply to the editor or an editorial material of a related paper might be separately indexed in the database and the match between a CV publication and this type of documents can give the highest similarity score. Such pairs are responsible for the wrong matches having a high Salton similarity score (higher than 0.65). This group can be seen in Figure 4b marked as class 1. In this group 3 wrong matches stand out with their quite low Levenshtein similarity score. The cause of this for two of them is having a lot of co-authors, and their sequence is not the same as that in the related WoS publications' bibliographic description.

For the other wrong match, the author mentions that the related issue is a special issue and gives the subject of this special issue. But the same author has another paper the title of which is the same with the subject of the special issue. As a result, the Salton score for the CV reference and this wrong WoS publication pair took the highest value. However, the

maximum Levenshtein score for this pair is 0.39, which is quite low. When we checked the Levenshtein score for the correct pair, we observed that it is 0.54. In other words, even when a high Salton score is obtained, it is also better to double check using the Levenshtein score.

The second group, indexed as class 2 in the Figure 4b, comprises only one pair. The related papers are written by the same co-author(s) and the titles are identical. However, the paper reported in the CV is published in a different journal. When we checked the correct WoS publication according to journal name, issue and begin-end pages, we observed that the title is shorter. As a result, another pair is retrieved due to the title length. It seems that the author made a mistake in his/her CV. This kind of errors is out of the scope of this study since we assume that the information given in the CV is correct.

The third group in the Figure 4b includes the pairs having the authors of the top match publications are not relevant. As an example, one author gives also editor names for the related source. However those editors are the co-authors of another publication in the same source. Therefore, more common 3-grams would be found and prevented the correct match from appearing on the top.

The last group of wrong matches in Figure 4b is class 4; this comprises the publications which are written in the field of high energy physics and coming from the same author's CV. There are 10 publications in this group and each of them has more than 200 co-authors, which is quite common in this field. The author in his/her CV gives the name of the corporation (team) instead of giving all individual co-author names. As a result, a very low Salton score is obtained since all co-authors are included when WoS publications are built. As a result, the number of unique 3-grams inflates on the WoS part and this directly decreases the Salton measure. The incorrect matches in such cases are obviously inevitable and the WoS publications with so many co-authors should be matched and assigned with utmost caution.

According to our observations in the context of these wrong matches, we can draw some important conclusions. A micro-level, notably an individual level bibliometric exercise requires an extremely careful assessment procedure (Glänzel and Wouters, 2013). For group 1 in Figure 4b, the pairs provide indirect but still useful information and they might be considered as tolerable depending on the user's aim. When considering the rest of the pairs in the three remaining groups in Figure 4b and their highest similarity scores, intuitive and empirical thresholds might be suggested. The pairs having a Salton score higher than 0.65 and having a Levenshtein distance higher than 0.50 seem to give correct WoS matches with quite low error rates. This means that the pairs lower than these values need to be checked manually. This area is shown by vertical and horizontal lines in Figure 4.

As to meso- and macro-level applications, we can implement the suggested method more flexibly. If we remember the wrong match groups again, the pairs belonging to groups 3 and 4, unlike that in group 2, have the maximum similarity scores with the related WoS publications whose author(s) is totally unrelated. Those pairs refer to different publications by the same authors. This means that corporate information may still correctly match. We can again suggest a range for the manual cleaning considering the highest wrong match score in group 3, that is, the pairs having a Salton and Levenshtein score lower than 0.52 and 0.42, respectively. When checking this area in Figure 4, one can observe that the amount of manual cleaning seems to become quite low.

Finally, we can conclude that a pair having both Salton and Levenshtein score lower than 0.40 indicate that the related CV is very likely not to be indexed in the WoS.

Conclusion

A set of 6.520 references, that are known to be indexed in the WoS, were taken from CVs provided by various authors. Salton's measure based on overlapping 3-grams was then used to link them to their counterparts in the WoS database. For each CV reference, a WoS

publication is retained as a correct match according to the highest obtained Salton similarity score. It was observed that 99% of the matches was correct when comparing them with the results of a manual cleaning and assignment process. This cleaning and searching process usually requires a severe manual effort. With the proposed retrieval model given in the information-retrieval phase, we suggest that it is possible to find the most related WoS publication automatically and this may help to significantly decrease the amount of manual effort.

To check, confirm and complement the Salton score, we applied Kondrak's (2005) method, which uses also the positions of N-grams besides common N-grams, for each CV–WoS pair. This suplementary measure proved to be quite consistent with the Salton measure. A detailed analysis of the incorrect matches was given and some thresholds were suggested for manual cleaning in order to be implemented in bibliometric studies at different levels. The suggested method has a potential to give more robust results for the meso and macro level bibliometric applications. Of course, the tolerance to the mistakes depends on the application. Extracting content-relevant information from publication strings, could possibly improve the accuracy of matching. Eventually, the method applied in this paper can especially be leveraged for the evaluation purposes such as job promotion or institutional evaluation cases at micro or meso level assessment studies, where large number of CVs is to be dealt with.

Acknowledgment

The authors would like to thank Prof. Wolfgang Glänzel for the valuable comments and remarks**.**

References

Abou-Assaleh, T., Cercone, N., Keselj, V., & Sweidan, R. (2004). Detection of New Malicious Code Using N-grams Signatures. *In PST* , 193-196.

Adams, E. S., & Meltzer, A. C. (1993). Trigrams as index element in full text retrieval: observations and experimental results. *In Proceedings of the 1993 ACM conference on Computer scienc*e, 433-439.

Cavnar, W. B. (1993). N-gram-based text filtering for TREC-2. *Ann Arbor*, 1001, 48113-4001.

Cavnar, W.B. (1994). Using an N-gram-based document representation with a vector processing retrieval model. *In: Harman DK, Ed. Proceedings of the Third Text Retrieval Conference (TREC-3), NIST Special Publication* 500-226, pp. 269–278.

Cavnar, W.B. & Trenkle, J.M. (1994). N-Gram-Based Text Categorization. *In: Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, US*, 161–175.

Cohen, J.D. (1995). Highlights: Language- and domain-independent automatic indexing terms for abstracting. *Journal of the American Society for Information Science,* 46, 162–174.

Comlekoglu, F.M. (1990). Optimizing a text retrieval system utilizing N-gram indexing. *Ph.D Thesis, George Washington University.*

Damashek, M. (1995). Gauging similarity with n-grams: Language-independent categorization of text. *Science*, 267, 843–848.

Giles, C.L., Bollacker, K.D., Lawrence, S. (1998). CiteSeer: an automatic citation indexing system. *Digital 98 Libraries. Third ACM Conference on Digital Libraries*, 89-98.

Glänzel, W. & Schoepflin, U. (1994). Little scientometrics, big scientometrics... and beyond?. *Scientometrics* 30(2) , 375-384.

Glänzel, W. and Czerwon, H.J. (1996). A new methodological approach to bibliographic coupling and its application to the national, regional and institutional level. *Scientometrics* 37(2), 195-221.

Glänzel, W., Wouters, P. (2013). The dos and don'ts in individual-level bibliometric. *Plenary session at the 14th International Conference on Scientometrics and Informetrics, Vienna, Austria.*

Goller, C., Löning, J., Will, T., & Wolff, W. (2000). Automatic Document Classification-A thorough Evaluation of various Methods. *ISI*, 145-162.

Gong, C., Huang, Y., Cheng, X., & Bai, S. (2008). Detecting near-duplicates in large-scale short text databases. *In Advances in Knowledge Discovery and Data Mining*, 877-883, Springer Berlin Heidelberg.

Kanaris, I., Kanaris, K., Houvardas, I., Stamatatos, E. (2007). Words versus character n-grams for anti-spam filtering. *International Journal on Artificial Intelligence Tools* 16(6) , 1047-1067.

Kešelj, V., Peng, F., Cercone, N., & Thomas, C. (2003, August). N-gram-based author profiles for authorship attribution. *In Proceedings of the conference pacific association for computational linguistics, PACLING*, 3, 255-264.

Khattree , R. & Naik, D.N. (2000). Multivariate Data Reduction and Discrimination With SAS Software. *Cary, NC: SAS Institute Inc. Wiley.*

Kondrak, G. (2005). N-gram similarity and distance. *Proceedings of the Twelfth International Conference on String Processing and Information Retrieval (SPIRE 2005)*, 115-126, Buenos Aires, Argentina.

Levenshtein, V.I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics-Doklady*, 10, 707-710.

Mihalcea, R. and Nastase, V. (2002). Letter level learning for language independent diacritics restoration. *In: Proceedings of the 6th Conference on Natural Language Learning (CoNLL-2002),* 105–111.

Shawe-Taylor, J. & Cristianini, N. (2000). Kernel Methods for Pattern Analysis. *Cambridge Univ. Press.*

SAS Institute Inc. (2008). *SAS/STAT 9.2 User's Guide. Cary, NC: SAS Institute Inc.*

Teufel, B. (1988). Natural language documents: Indexing and retrieval in an information system. *In: Proceedings of the 9th International Conference on Information Systems, Minneapolis, Minnesota,* 193– 201.

Lawrence, S., Giles, C.L. and Bollacker, K.D. (1999). Autonomous Citation Matching. *In: Etzioni, O., Muller, J.P. and Bradshaw, J.M. eds. AGENTS '99. Proceedings of the Third Annual Conference on Autonomous Agents, May 1-5, 1999, Seattle, WA, USA. New York: ACM Press,* 392-393.

Larsen, B. (2004). References and Citations in Automatic Indexing and Retrieval Systems - Experiments with the Boomerang Effect. *PhD thesis, Royal School of Library and Information Science.*

Mcnamee, P. and Mayfield, J. (2004). Character n-gram tokenization for European language text retrieval. *Information Retrieval* 7.1-2, 73-97.

Piskorski, J., Sydow, M. (2007). String Distance Metrics for Reference Matching and Search Query Correction. *In : Abramowicz, W. (ed.) BIS 2007. LNCS, vol. 4439, pp. 353–365. Springer, eidelberg*, doi:10.1007/978-3-540-72035-5-27

Sen, S. K., & Gan, S. K. (1983). A mathematical extension of the idea of bibliographic coupling and its applications. *Annals of Library Science and Documentation*, 30(2), 78-82.

Salton, G. and McGill, M. (1983). Introduction to Modern Information Retrieval. *McGraw-Hill New York,* 1983.

Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal,* 27, 379–423 and 623–656.

Van Raan, A. FJ. (1997). Scientometrics: State-of-the-art. *Scientometrics,* 38(1), 205-218.

Zamora, E.M., Pollock, J.J. and Zamora, A. (1981). The use of trigram analysis for spelling error detection. *Information Processing and Management*, 17, 305–316.

FACULTY OF ECONOMICS AND BUSINESS
DEPARTMENT OF MANAGERIAL ECONOMICS, STRATEGY AND INNOVATION
Naamsestraat 69 bus 3500
3000 LEUVEN, BELGIË
tel. + 32 16 32 67 00
fax + 32 16 32 67 32
info@econ.kuleuven.be
www.econ.kuleuven.be/MSI